Coordinating batching decisions in manufacturing networks

van der Zee, D.J.

Published in:
International Journal of Production Research

DOI:
10.1080/00207543.2017.1317926

International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: http://www.tandfonline.com/loi/tprs20

Coordinating batching decisions in manufacturing networks

Durk-Jouke van der Zee

To cite this article: Durk-Jouke van der Zee (2017) Coordinating batching decisions in manufacturing networks, International Journal of Production Research, 55:18, 5405-5422, DOI: 10.1080/00207543.2017.1317926

To link to this article: http://dx.doi.org/10.1080/00207543.2017.1317926

Published online: 25 Apr 2017.

International Journal of Production Research, 2017
Vol. 55, No. 18, 5405–5422, https://doi.org/10.1080/00207543.2017.1317926

Coordinating batching decisions in manufacturing networks

Durk-Jouke van der Zee*

Faculty of Economics & Business, Department of Operations, University of Groningen, Groningen, The Netherlands

(*Received 31 October 2016; accepted 21 March 2017*)

Family-based dispatching heuristics aim for improving job flow times by reducing time spent on set-ups. They realise set-up efficiencies by batching similar types of jobs. By their intuitiveness and the simplicity of their decision logic, they may contribute to an easy to implement and viable strategy in many practical settings. Similar to common dispatching rules most existing family-based dispatching heuristics are myopic, i.e. their decision scope is restricted to a single manufacturing stage. Hence, they neglect opportunities for improving shop performance by coordinating batching decisions with other manufacturing stages. Case examples from industry underpin the need for exploring these opportunities. We do so by studying a simple two-stage flow shop, entailing a serial and a batch stage. To facilitate shop coordination we propose extensions to existing family-based dispatching heuristics. Extended heuristics seek to further increase set-up efficiencies by allowing for upstream job re-sequencing, and pro-active set-ups, i.e. set-ups that may be initiated prior to the arrival of a job. Outcomes of an extensive simulation study indicate significant performance gains for extended heuristics vs. existing heuristics. Performance gains are largest for moderate and high set-up to run-time ratios.

Keywords: batch processing; dispatching rules; manufacturing networks; simulation; shop floor control; sequencing

1. Introduction

Dispatching involving set-up times plays an important role in today's industry for the timely delivery of reliable products. The set-up process is not a value added factor, and hence, set-up times need to be explicitly considered while dispatching decisions are made in order to increase a firm's competitiveness in terms of productivity, eliminate waste, improve resource utilisation and meet deadlines (Allahverdi 2015; Bevilacqua et al. 2017). Family-based dispatching heuristics support decision-making by seeking to reduce time spent on set-up times by grouping (batching) and jointly dispatching jobs sharing similar requirements with respect to machine set-up.

In this article, we study the coordination of family-based dispatching decisions with upstream manufacturing stages. Our work is motivated by a case study concerning a manufacturer of centrifugal pumps for the (petro) chemical industry, horticular market and shipbuilding industry (Bokhorst, Nomden, and Slomp 2008; Nomden 2011; van der Zee, Gaalman, and Nomden 2011). Its mechanical processing department produces parts (casings, impellers, shafts etc.) for the various assembly cells. In total 10 part families, 14 types of cutting operations may be distinguished (for example, turning, milling, drilling etc.). The company is driven towards more product customisation, due to fierce price-based competition of mass producers and low-wage countries. Tight due dates, and the need to safeguard operations' efficiency – as part lot sizes decrease due to product customisation – make exploitation of set-up efficiencies in parts production an interesting and relevant issue for the company. Operators seek to increase set-up efficiencies by applying family-based dispatching heuristics, in deciding what (type of) job to produce next on a batch machine. Moreover, their batching decisions may be coordinated with dispatching decisions for upstream or downstream stages. In doing so, operators rely on their access to a computerised shop floor control system, and their own observations of shop status. By adjusting part sequencing at upstream stages – for increasing set-up efficiencies at a batch machine, and/or linking batching decisions to workload observed for downstream stages, operators aim for further flow time improvements. These practices clarify that use and benefits of family-based dispatching heuristics should be considered in a shop-wide context.

We observed how many manufacturers producing small batches or discrete parts encounter dilemma's similar to the one faced by the pump manufacturer. Some further case examples, covering various industries, concern the manufacturing of high tech defence products (Nomden 2011), production of metal plate components and assemblies for cars, power tools and household appliances (Nomden and van der Zee 2008), the operation of paint shops and press shops in car

*Email: d.j.van.der.zee@rug.nl

manufacturing (Salmasi, Logendran, and Skandari 2010), furniture production (Wilson, King, and Hodgson 2004) and production of aircraft engine blades (Li 1997). From a manager's point of view, it is relevant to understand how strategies for coordinating family-based dispatching decisions with upstream and downstream stages may contribute to operational responsiveness of the shop (Bevilacqua et al. 2017). Insights obtained inform his decision-making on enabling such coordination. Typically, successful implementation and operation of coordination mechanisms implies knowing and meeting various requirements on the content and workings of shop floor control systems, as well as the tasks and skills of their users, i.e. planners and/or operators, also compare the above case example.

In past years, family-based dispatching heuristics received significant attention in literature (Pickardt and Branke 2012). However, most studies consider the development of myopic heuristics that restrict decision scope to a single manufacturing stage, and rely on local information for underpinning dispatching decisions. Only few heuristics are proposed that seek to exploit shop floor data and control systems to a greater extent. Benefits of including forecasted or predicted future job arrivals in heuristic decision-making are considered by Kannan and Ghosh (1996), Mahmoodi and Martin (1997), Reddy and Narendran (2003) and Nomden, Van der Zee, and Slomp (2008). Furthermore, in a recent study van der Zee, Gaalman, and Nomden (2011) show how overall shop performance may benefit from considering work in process for downstream manufacturing stages in family based dispatching. Surprisingly, strategies for coordinating batching decisions with upstream stages receive no attention in literature.

Starting from the above observations, we distil a potential for improving existing family based heuristics by involving upstream stages in decision-making. In order to exclude other influencing factors, we chose to study a simple two-stage flow shop. The shop entails a serial and a batch stage. To enable coordination of batching decisions with sequencing decisions at the upstream serial stage, we extend existing family-based heuristics towards the shop level. Essentially, the extended heuristics build their dispatching decisions for either stage on the creation and assessment of a set of alternative shop schedules. Each schedule reflects job availability and progress at the batch stage and a unique job sequence at the serial stage. In building schedules, heuristics allow for the possibility of pro-active set-ups, i.e. set-ups that may be initiated in anticipation of jobs yet to arrive at the batch machine. Clearly, pro-active set-ups may reduce job waiting times. A simulation study is designed to demonstrate and analyse the potential of existing and extended heuristics in terms of mean flow times per job.

The paper is structured as follows. In Section 2, we review existing heuristics for family-based dispatching. Next, in Section 3, the shop environment and the decision structure for shop control are described in detail. In Section 4, we propose extensions to existing heuristics. Existing and extended heuristics are evaluated by a simulation study (Sections 5 and 6). Finally, main conclusions are summarised in Section 7.

2. Literature review

Family-based dispatching heuristics build on Group Technology (GT) principles by stressing the exploitation of job similarities in shop operation and control (Nomden, Slomp, and Suresh 2006). By grouping jobs with similar needs with respect to machine set-up for joint dispatching, they seek to lower set-up frequencies. Gains obtained impact both the shop service level, by contributing to timeliness of its operations – as job flow times may be reduced, and operational costs – as less efforts are involved in readying machines for producing parts. Note that in many cases set-up times should not be used as a proxy for set-up costs in the modelling of scheduling operations, due to the characteristics of the underlying cost-function (Ciavotta, Meloni, and Pranzo 2013). Similar to common dispatching heuristics, family-based dispatching heuristics support prompt decision-making on the job to be processed next. Job selection builds on intuitive decision logic, and relies on limited, usually local, information (McKay, Safayeni, and Buzacott 1988; Stuber 1998; Oeulhadj and Petrovic 2009). As such they may contribute to an easy to implement and viable strategy for shop control in many practical settings, in which size and complexity of the scheduling problem, and/or lack of scheduling software hinder analytic solutions (Sels, Gheysen, and Vanhoucke 2012; Allahverdi 2015).

Below we will classify family-based dispatching heuristics, and discuss their decision structure. Next, we will review those heuristics that go beyond the notion of local, single stage optimisation of batch processes. We consider heuristics adopting the minimisation of mean flow time as an objective. Relevance of minimising flow times is related to, among others, improvement of customer responsiveness, maintaining flexibility, improvement of product quality, less need for relying on forecasts, reducing costs associated with work in process, and making better forecasts (Hopp and Spearman 2008; Pinedo 2015). See Mosier, Elvers, and Kelly (1984), Mahmoodi and Dooley (1991), Ponnambalam, Aravindan, and Reddy (1999), Pickardt and Branke (2012) and Neufeld, Gupta, and Buscher (2016) for heuristics addressing due date related criterions.

Family-based dispatching heuristics may be classified in three categories (Pickardt and Branke 2012): purely set-up oriented heuristics, composite heuristics and (purely) family-based heuristics. Differences among categories relate to their choice of objective, and their decision structure. *Purely set-up oriented heuristics* strive to minimise machine time spent on set-ups. A well-known example is the Short Set-up Time rule (SST), prioritising the job requiring least set-up time (Gavett 1965). In general, purely set-up oriented heuristics will not suffice in case of a flow time criterion due to the fact that job processing times are not considered in job priority setting (Pickardt and Branke 2012). For that reason we will not discuss this category here.

Composite heuristics and *purely family-based heuristics* differ from each other by their choice of process batch size. While former heuristics assume dispatching decisions to be restricted to the next job, the latter heuristics allow for the joint dispatch of multiple jobs belonging to the same job family. This is reflected in their decision structure. Whereas composite heuristics make a single decision on the job to process next – building on a priority index, purely family-based heuristics are typified by three ordered decisions in determining the composition of the process batch to be produced next (Mosier, Elvers, and Kelly 1984):

(a) Decision moment: When to select a new family of jobs for servicing.
(b) Family type selection: Which of the families to process next – assuming the decision in (a) has been made.
(c) Job sequencing: Which job is to be selected from the chosen family.

A well-performing *composite heuristic* is the Shortest Normalised Setup and Processing Time heuristic (SNSPT), as proposed by Kochhar and Morris (1987) for controlling a flexible flow line. The associated priority index, i.e. $\alpha\frac{s_j^B}{\hat{s}}+\beta Q\frac{p_{i,j}^B}{\hat{p}}$, sums processing time for job i in family j ($p_{i,j}$) and family set-up time (s_j), after being (1) divided by their respective mean values in the queue ($\hat{p}$, $\hat{s}$) and (2) weighted (α, β, Q).Q is –1 if the input buffer of the next workstation is more than γ% full and +1 otherwise. For details on tuning α, β, γ see Kochhar and Morris (1987). Pickardt and Branke (2012) found that SNSPT outperforms alternative composite heuristics, given a specific setting of weights. Also it performs well relative to a subset of purely family-based heuristics.

In past years several *purely family-based heuristics* have been proposed. Heuristics differ from each other by their implementation of the decision structure, see above. As far as the choice of *decision moment* (a) is concerned, two main subclasses may be distinguished, i.e. *exhaustive* and *non-exhaustive heuristics* (Mahmoodi and Dooley 1991). Whereas exhaustive heuristics only allow decision-making at the moment the queue of jobs for the current family is exhausted, non-exhaustive heuristics specify a truncation-mechanism for deciding on the switching of families. Various truncation mechanisms have been studied that link a decision moment to restrictions on batch size (Mosier, Elvers, and Kelly 1984; Ruben, Mosier, and Mahmoodi 1993; van der Zee 2010, 2015) or a specified time lapse (Russell and Philipoom 1991). Despite a long-lasting debate in literature on the benefits of either subclass – exhaustive or non-exhaustive – no single heuristic has been proposed indicating overall best performance. Shop characteristics tend to be decisive in determining the right choice of heuristic (van der Zee 2010; Pickardt and Branke 2012). For example, the non-exhaustive MASP_AD and MASP_HY heuristics may outperform exhaustive heuristics in case of moderate and high variances of processing and set-up times. In turn, the exhaustive MASP (Russell and Philipoom 1991) and MAS (Nomden, Van der Zee, and Slomp 2008) heuristics indicate good performance for low variances of processing and set-up times (van der Zee 2010).

At each decision moment purely family-based dispatching heuristics employ a priority index for *family type selection* (b). Whereas early heuristics employ common dispatching rules like First Come First Serve (FCFS) or Shortest Processing Time (SPT) for setting family priority, more recent heuristics adopt a more informed index relating to the Weighted Shortest Processing Time rule. MASP (Russell and Philipoom 1991) and MASP_AD provide important and well-performing examples of the latter heuristics. The exhaustive MASP heuristic computes workload by summing family set-up time and processing times of those jobs in the process batch, while setting weight equal to batch size, i.e. family queue length. The non-exhaustive MASP_AD heuristic extends MASP by allowing for alternative choices of batch size.

Finally, purely family-based dispatching heuristics adopt common dispatching rules like FCFS, SPT or EDD for *sequencing jobs* (c) in the process batch. See Blackstone, Phillips, and Hogg (1982), Jayamohan and Rajendran (2000), Mizrak and Bayhan (2006), and Sarin, Varadarajan, and Wang (2011) for overviews.

Aforementioned heuristics may be typified as myopic, as they relate decision-making to a single manufacturing stage, while relying on local information only. Many authors show, how respective heuristics, despite their myopic nature, may contribute significantly to shop performance. At the same time, both practice (compare Section 1) and literature (for example, Oeulhadj and Petrovic 2009) hint at opportunities for improving heuristics, by suggesting good use of shop data and exploiting shop floor control systems, in coordinating batching decisions with upstream and downstream stages.

So far, few heuristics are proposed that consider exploiting shop floor control data beyond those available for the batch stage. VCM3 and VCM5 (Kannan and Ghosh 1996) start from the idea that the operator is informed about those job families which, next to having one or more jobs in the queue, also have jobs being processed in an upstream manufacturing stage. VCM3 assigns priority to the respective set of families in family selection. VCM5 restricts focus to the current family being processed. The authors only report minor performance improvements relative to none look-ahead heuristics, which may be attributed to the rather coarse manner in which future arrivals are accounted for. The LPTMM heuristic (Mahmoodi and Martin 1997) predicts future arrivals using historical data on demand rates, and uses this information to improve family priority setting. It shows favourable performance in situations with an oscillating demand rate. Reddy and Narendran (2003) extend the work of Mahmoodi and Martin. Their PH heuristic prioritises job families for which the preset economic lot size is met earliest. Reddy and Narendran found how their heuristic performed well for configurations with a high shop workload and high set-up to run-time ratio. Nomden, Van der Zee, and Slomp (2008) show how shop performance may benefit significantly from the inclusion of forecast data on future arrivals in family priority setting. Finally, van der Zee, Gaalman, and Nomden (2011) show how shop-wide performance may benefit from including information on current workload for downstream machines in family-based dispatching.

Essentially, aforementioned heuristics use data on shop status beyond local information in an attempt to improve shop performance. In this article, we build on this work by proposing further extensions to existing heuristics that aim to coordinate dispatching decisions for the batch stage with those for upstream stages. A first extension concerns the possibility to re-sequence jobs at upstream machines. By allowing for job re-sequencing prior to their arrival at the batch stage further options for increasing set-up efficiencies at the batch stage may be exploited. Secondly, extended heuristics allow for pro-active set-ups, i.e. set-ups that may be initiated prior to job arrival at the batch stage. Heuristics development and testing concerns a two-stage flow shop. ‘Simplicity’ of the shop structure is meant to foster understanding of contributions made by heuristics’ extensions. Insights obtained on heuristics’ construction and potential are meant to support further research on heuristics addressing more complex multi-stage systems, and clarify the need for doing so. Clearly, our choice of decision scope for the extended heuristics somewhat confines their direct use for practical applications. We remark, however, that many two-stage manufacturing systems can be found in industry (Groover 2008; Ciavotta, Meloni, and Pranzo 2013), or can be individuated as such by a model aggregation or simplification procedure.

3. Shop description and decision structure

3.1 *Shop description*

We consider a two-stage flow shop, see Figure 1. An overview of the notation can be found in Appendix 1. Manufacturing stages concern a serial machine (S), and a batch machine (B), respectively. Buffers are used to store incoming jobs and to decouple both stages. For all buffers unlimited storage capacity is assumed. Each job belongs to a certain family $j \in J$. For each family $j \in J$, it is distinguished between two sets of jobs $I_j^S(t_0^S)$, $I_j^B(t_0^B)$ referring to jobs available in

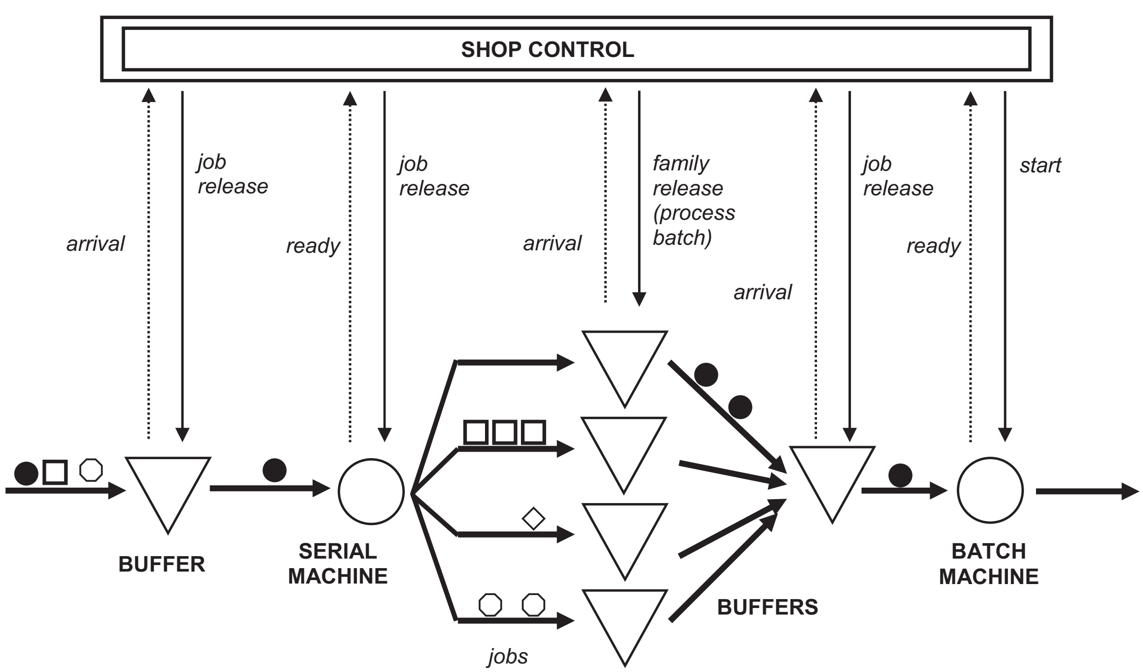

Figure 1. Shop lay-out.

queue at the serial machine and the batch machine, respectively, at a decision moment (t_0^S, t_0^B), i.e. a moment the operator may decide to switch families. Each job is processed at the serial machine for an amount of time $(p_{i,j}^S)$, with $i \in I_j^S(t_0^S)$. Total number of jobs in queue at the serial and batch machine for each family j equal q_j^S and q_j^B, respectively. Each job family requires a specific set-up at the batch machine. This so-called major set-up is associated with a set-up time $s_{j_0,j}^B$. Length of the set-up time is determined by the current set-up – for family j_0 – and the required set-up for family j. Obviously, $s_{j_0,j}^B = 0$ for $j = j_0$. Note that the definition of set-up times allows for the presence of both sequence dependent and sequence independent set-ups. Job-related, so-called minor set-ups, are assumed to be included in job processing times $(p_{i,j}^B)$, with $i \in I_j^B(t_0)$. For each family, jobs in queue at both stages are assumed to be ordered according to their processing times, i.e. $p_{1,j}^S < p_{2,j}^S < \cdots < p_{n_j^S,j}^S$, with $n_j^S = \left|I_j^S(t_0^S)\right| = q_j^S$, and $p_{1,j}^B < p_{2,j}^B < \cdots < p_{n_j^B,j}^B$, with $n_j^B = \left|I_j^B(t_0^B)\right| = q_j^B$.

3.2 *Decision structure for shop control – decoupling dispatching decisions*

In this section, we elaborate on the decision structure for shop control. The shop control concerns both stages of the shop, see Figure 1. As an objective we consider the minimisation of mean shop flow time per job in the long run. Given N processed jobs, mean flow time per job $(\bar{T})$ is defined as:

$$\bar{T} = \frac{\sum_{j \in J} \sum_{i=1,2,\ldots} ft_{i,j}}{N}$$
$$\text{with}$$
$$ft_{i,j} = w_{i,j}^S + p_{i,j}^S + w_{i,j}^B + p_{i,j}^B \tag{1}$$
$$N = \sum_{j \in J} \sum_{i=1,2,\ldots} 1$$

In computing flow time for a job i belonging to family $j\,(ft_{i,j})$ we distinguish between waiting times $(w_{i,j}^S, w_{i,j}^B)$ and job processing times $(p_{i,j}^S, p_{i,j}^B)$ for the first (serial) stage and the second (batch) stage, respectively. Note how waiting time for the batch stage $(w_{i,j}^B)$ includes set-up times $(s_{j_0,j}^B)$.

For reasons of simplicity and clarity of understanding, in this section, we assume dispatching decisions for both stages to be decoupled, i.e. not coordinated. Dispatching decisions for the serial stage rely on a common dispatching rule for job sequencing, i.e. SPT. Existing family-based dispatching heuristics are employed in controlling the batch stage. In Section 4, we modify this decision structure by allowing for extended heuristics that coordinate batching decisions with the upstream serial stage. As a precursor to the introduction of the extended heuristics, we typify existing family-based heuristics by considering implementation issues. In line with Mosier et al. (1984) family-based dispatching heuristics are characterised by three ordered decisions in determining the process batch that is to be dispatched next. Decision logic for composite heuristics could be defined in accordance with this structure, by assuming them to be a specific class of purely family-based dispatching heuristics for which process batch size equals 1:

(a) Decision moment: When to select a new family of jobs for servicing.

In principle, a new family may be selected at each moment status of the batch machine changes by either a job arrival or a job being completed. Family-based dispatching heuristics, however, tend to restrict decision moments (t_0^B). *Exhaustive* heuristics only allow for a new family to be dispatched if the queue for the current family is empty. By maximising batch size in this way, they strive to increase set-up efficiencies. Non-exhaustive heuristics apply other type of restrictions that influence process batch size, and – hence – the choice of decision moments. Such restrictions may be related to, for example, time fences for processing a specific family (Russell and Philipoom 1991), or process batch sizes (Mosier et al. 1984, van der Zee 2010). Note that composite heuristics concern a specific class of non-exhaustive heuristics, for which process batch size is restricted to 1.

(b) Building the schedule: Family type selection – Which of the families to process next.

Family-based dispatching heuristics adopt a priority index for selecting the family to process next. Priority indices developed so far may be separated in two generations. A first generation relates family priority setting to the application of common dispatching rules like FCFS and SPT. For example, the so-called FCFAM heuristic (Flynn 1987) builds on the logic of the FCFS rule. It prioritises families by considering the *earliest arrival moment* at the batch machine $(t_{i,j}^B)$ among the jobs i available for a family $j \in J$ at the decision moment (t_0^B):

$$j^*_{FCFAM} = \underset{i \in I^B_j(t^B_0); j \in J}{\arg\min}\ t^B_{i,j} \quad (2)$$

In turn, the more recent exhaustive MASP heuristic (Russell and Philipoom 1991) implements the concept of the well-known Weighted Shortest Processing Time rule (Pinedo 2015) in prioritising families:

$$j^*_{MASP} = \underset{j \in J; q^B_j > 0}{\arg\min} \frac{s^B_{j_0,j} + \sum_{i=1}^{q^B_j} p^B_{i,j}}{q^B_j} \quad (3)$$

MASP selects the family j^* for which a minimum weighted workload is foreseen for servicing. It estimates workload by the sum of family set-up time ($s^B_{j_0,j}$) and cumulative processing times $\left(\sum_{i=1}^{q^B_j} p^B_{i,j}\right)$ for those jobs within family j that are available at the decision moment, while choice of weights is related to family queue length (q^B_j). Other exhaustive heuristics differ from MASP by choosing alternative definitions of workload and/or weight, see Nomden, Van der Zee, and Slomp (2008) for an overview. Furthermore, non-exhaustive heuristics may couple family type selection and the choice of process batch contents. For example, MASP_AD (van der Zee 2010) allows for alternative choices of weight, i.e. batch size, in selecting a new family, and determining batch size.

(c) Job sequencing: Which job is to be selected from the chosen family.

Job sequencing is realised by employing *common dispatching rules*, such as, for example, FCFS and SPT.

4. Coordinating batching decisions with upstream stages

In this section, we propose extended heuristics that coordinate batching decisions for a two-stage flow shop. Shop coordination is linked to (i) the possibility to concert batching decisions with sequencing decisions at the upstream serial machine, and (ii) the possibility of pro-active set-ups. Below we first sketch implementation of heuristics' extensions in broad terms. Next, we discuss implementation details, starting from the decision structure for shop control introduced in Section 3. Existing heuristics considered are FCFAM (Flynn 1987), MASP (Russell and Philipoom 1991), MAS (Nomden, Van der Zee, and Slomp 2008), MASP_AD (van der Zee 2010) and SNSPT (Kochhar and Morris 1987). FCFAM is often used in literature as a benchmark for establishing potential performance gains for more informed family-based dispatching heuristics. Our choice of remainder heuristics is motivated by their good performance in previous studies. In the absence of a single best heuristic (Pickardt and Branke 2012) together they are meant to provide good 'coverage' for a wide range of shop configurations. An overview of the heuristics introduced in this section, can be found in Appendix 2.

Essentially, coordination of dispatching decisions for both shop stages implies the need for building a joint shop schedule. To facilitate shop scheduling we propose to:

- Synchronise *decision moments*: Dispatching decisions for both stages of the flow shop are concerted by providing a new shop schedule at the moment a next job is to be dispatched at the serial stage.
- Adjust *procedures for building the shop schedule*: Instead of decoupling sequencing decisions at the serial stage and dispatching decisions at the batch stage (compare Section 3), extended heuristics build and assess alternative shop schedules that include all jobs available at the batch stage at the decision moment, as well as the job that will be dispatched at the serial stage.

Let us now consider implementation details of proposed heuristics' extensions. Table 1 provides an overview of proposed heuristics. Heuristics' names reflect the name of the existing heuristic they build on. Heuristics' extensions relative to the underlying existing heuristics are marked by adding '_C_P' to their names, also see above and Section 5. Similar to existing heuristics, extended heuristics are typified by a three step approach (compare Section 3):

(a) Decision moment: When to create a shop schedule

A new shop schedule $S*$ is created at the moment status of the serial stage changes (t^S_0). Serial machine status may change due to a job being completed or a new job arrival (in case the serial machine has starved). Schedule $S*$ includes the next job $c*$ to be dispatched at the serial machine, and all jobs available at the batch stage at the decision moment (t^S_0). Execution of $S*$ is triggered at the decision moment (t^S_0) and the moments (t^B_0) the batch machine completes service for a process batch served or jobs arrive at the batch stage (in case the batch machine has starved). Note that the above choice of decision moments suggests a decoupling of the scheduling activity, and schedule implementation for the batch stage, also see above.

Table 1. Extended family based dispatching heuristics.

Heuristic	Decision Moments and Family Switching Strategy			Priority Indices for Shop Schedule Creation		Job Sequencing
	Decision Moments	Exhaustive	Batch Size	Family sequencing for batch processing	Job selection for serial processing	
FCFAM_C_P	t_0^S, t_0^B	Yes	$k = q_j^B$	$t_{i,j}^B$	$d(S_c, c)$	SPT
MASP_C_P	t_0^S, t_0^B	Yes	$k = q_j^B$	$\frac{\tilde{s}_{j_0,j}^B + \sum_{i=1}^k p_{i,j}^B}{k}$	$d(S_c, c)$	SPT
MAS_C_P	t_0^S, t_0^B	Yes	$k = q_j^B$	$\frac{\tilde{s}_{j_0,j}^B}{k}$	$d(S_c, c)$	SPT
MASP_AD_C_P	t_0^S, t_0^B	No	$k = 1...q_j^B$	$\frac{\tilde{s}_{j_0,j}^B + \sum_{i=1}^k p_{i,j}^B}{k}$	$d(S_c, c)$	SPT
SNSPT_C_P	t_0^S, t_0^B	No	$k = 1$	$\alpha \frac{\widetilde{s}_{j_0,j}^B}{\bar{s}} + \beta Q \frac{p_{i,j}^B}{\bar{p}}$ $\alpha = \beta = Q = 1$	$d(S_c, c)$	SPT

(b) Build the shop schedule: Coordinate sequencing and batching decisions

A stepwise procedure is employed for establishing the next job c^* to be processed at the serial stage, and a new schedule S^* for the batch stage, simultaneously:

(1) *Determine the set of candidate jobs* (C) *for dispatching at the serial stage*: A two phase approach is adopted to compose C. First, build an initial set of candidate jobs by selecting the first job in queue for each family $j \in J$, i.e. the job requiring the shortest processing time at the serial stage, compare Section 3. Next, for each job $c \in C$ check the present schedule for the batch stage for the first possible moment ($st^B(c)$) it could be processed, i.e. the moment the machine completes service for the process batch that is current upon its arrival moment ($t_c^B = t_0^S + p_{c,j}^S$). If the respective job has to wait ($t_c^B < st^B(c)$), allow for a replacement by a 'better' candidate job in queue belonging to the same family that would (i) arrive before $st^B(c)$, but (ii) requires less processing time at the batch stage than c.

(2) *Build a set of alternative shop schedules* $\bar{S}$, starting from the set of candidate jobs (C): For each alternative choice of job $c \in C$ determine jobs that would be in queue at the batch stage upon its arrival – after being processed at the serial stage. Next, build a schedule $S_c \in \bar{S}$ by sequencing process batches, thereby employing the family priority index (see Table 1). Note that schedules built by exhaustive heuristics assume family incidence to be one at most, while non-exhaustive heuristics allow for multiple process batches to be composed for each family. Family priority indices acknowledge the possibility of pro-active set-ups by adjusting set-up times in family priority setting by only considering set-up time left (if any) at the job arrival moment (t_c^B), i.e.:

$$\tilde{s}_{j_0,j}^B = \begin{cases} \max(0, s_{j_0,j}^B - (t_c^B - T^B)) & \text{if } t_c^B > T^B \\ s_{j_0,j}^B & else \end{cases} \tag{4}$$

Equation (4) restricts pro-active set-ups to those situations in which the batch machine has starved at a moment (T^B) prior to the arrival moment of the selected job at the batch machine ($t_c^B = t_0^S + p_{c,j}^S$).

A somewhat different strategy is applied in building the schedule for those cases the selected job c would arrive at the batch machine at the moment the family it belongs to is current. Outcomes of the strategy depend on the choice of heuristics applied for shop control, being either exhaustive or non-exhaustive. In the former case the respective job c is added to the current process batch, implying no further changes to the current schedule. Non-exhaustive strategies, however, only allow the possibility of letting the respective job c replace one of the jobs within the current process batch, thereby seeking to minimise flow time for the process batch. Next, in accordance with the above procedure, they build a schedule for remainder jobs in queue at the batch stage, including the replaced job.

(1) *Assess decision options*: Determine priorities for all candidate jobs $c \in C$ by establishing the moment ($d_c = d(S_c, c)$) they will be dispatched according to corresponding schedules $S_c \in \bar{S}$. Best choice of job (c^*) is associated with the earliest dispatch moment at the batch stage. In case of a tie, i.e. the earliest dispatch moment coincides for multiple selected jobs, choose the job for which required set-up time and processing time

$(\tilde{s}^B_{j_0,j} + p_{c,j})$ are less. If this still results in a tie, a random choice is made among respective jobs. Given c^* the choice of S^* is straightforward, i.e. $S^* = S_{c^*}$. Note that job preferences built on the intuition that jobs favoured most in batch formation – as indicated by their place in line – make the largest contribution to shop performance as they increase set-up efficiencies most.

(2) *Implement dispatching decisions*: Replace the current shop schedule with the best schedule S^*. Next, dispatch the preferred job c^* at the serial stage.

(c) Job sequencing at the batch stage: Which job is to be selected from the chosen family

In line with previous research (Mahmoodi, Dooley, and Starr 1990; Wemmerlöv 1992; Pickardt and Branke 2012) all heuristics adopt the SPT rule for job sequencing within process batches at the batch stage.

5. Design of the simulation study

5.1 *Experimental design*

By our simulation study, we aim to gain insights in the added value of the shop coordination mechanisms implemented for the extended heuristics. The experimental design for the simulation study is shown in Table 2. Choice of fixed factors, experimental factors and their ranges are in line with and motivated by previous research, see, for example,

Table 2. Simulation study – overview of fixed and experimental factors.

Fixed factors							
Family mix					Equal share per family		
Inter-arrival time distribution					Exp		
Processing time distribution serial machine					Exp		
Workload serial machine					85%		
Set-up time distribution batch machine					Fixed set-up times (set-up matrix[a])		
Processing time distribution batch machine					Exp (Mean = 100)		
Dispatching rule for serial stage					SPT		
Experimental factors							
Number of job families (F)					4;8;16		
Set-up to runtime ratio (S/R)[b]					0.25; 0.50;1.00		
Workload batch machine (WL)					60%; 75%; 90%		
Mean job inter-arrival times[c]							
					Number of families		
					4	8	16
	Work load	60%	S/R	0.25	194	2	204
				0.50	152	158	162
				1.00	121	127	131
		75%	S/R	0.25	222	235	242
				0.50	172	183	191
				1.00	134	144	153
		90%	S/R	0.25	278	304	318
				0.50	213	236	250
				1.00	160	182	197
Heuristics (H)							
No shop coordination (HE = none)	Shop coordination (HE = C)				Shop coordination, pro-active set-ups (HE = C_P)		
FCFAM	FCFAM_C				FCFAM_C_P		
MASP	MASP_C				MASP_C_P		
MAS	MAS_C				MAS_C_P		
MASP_AD	MASP_AD_C				MASP_C_P		
SNSPT	SNSPT_C				SNSPT_C_P		

[a]See Table 3.
[b]Mean set-up time divided by mean processing time.
[c]Mean job arrival intervals have been determined for alternative settings of the number of families, set-up to runtime ratio and work load, assuming FCFAM is chosen as the family based dispatching heuristic.

Table 3. Simulation study – set-up matrix (16 families, S/R = 1).

	From family															
To family	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	80	120	160	40	80	120	160	40	80	120	160	40	80	120	160
2	40	0	120	160	40	80	120	160	40	80	120	160	40	80	120	160
3	40	80	0	160	40	80	120	160	40	80	120	160	40	80	120	160
4	40	80	120	0	40	80	120	160	40	80	120	160	40	80	120	160
5	40	80	120	160	0	80	120	160	40	80	120	160	40	80	120	160
6	40	80	120	160	40	0	120	160	40	80	120	160	40	80	120	160
7	40	80	120	160	40	80	0	160	40	80	120	160	40	80	120	160
8	40	80	120	160	40	80	120	0	40	80	120	160	40	80	120	160
9	40	80	120	160	40	80	120	160	0	80	120	160	40	80	120	160
10	40	80	120	160	40	80	120	160	40	0	120	160	40	80	120	160
11	40	80	120	160	40	80	120	160	40	80	0	160	40	80	120	160
12	40	80	120	160	40	80	120	160	40	80	120	0	40	80	120	160
13	40	80	120	160	40	80	120	160	40	80	120	160	0	80	120	160
14	40	80	120	160	40	80	120	160	40	80	120	160	40	0	120	160
15	40	80	120	160	40	80	120	160	40	80	120	160	40	80	0	160
16	40	80	120	160	40	80	120	160	40	80	120	160	40	80	120	0

Wemmerlöv and Vakharia (1991), Wemmerlöv (1992), Frazier (1996), Shambu, Suresh, and Pegels (1996), Jensen, Malhotra, and Philipoom (1996, 1998), Kannan (1998), Marsh, Shafer, and Meredith (1999), Chern and Liu (2003), Nomden, van der Zee, and Slomp (2008), and Pickardt and Branke (2012).

All configurations studied concern a two-stage flow shop. Job families are assumed to have an equal share in product mix. Job inter-arrival times are modelled by a negative exponential distribution. Job processing times are drawn from a negative exponential distribution. Mean processing time for the batch machine equals 100. Mean processing times for the serial machine are adjusted to job arrival rates in order to guarantee a workload of 85%. Set-up times required at the batch machine are specified by a set-up matrix, see Table 3. The serial stage is controlled according to an SPT rule.

Main experimental factor concerns the choice of heuristic for addressing the batch stage. Three categories of heuristics are considered. Firstly, existing heuristics (FCFAM, MASP, MAS, MASP_AD, SNSPT) are employed to reflect a shop setting for which no coordination of batching decisions is considered (HE = None). In turn, a coordinated approach towards shop control, encompassing both stages, is implemented by choosing among one of the extended heuristics, compare Section 4. To isolate and assess benefits of considering pro-active set-ups in family-based dispatching for shop performance we distinguish among extended heuristics not acknowledging pro-active set-ups and those that do (HE = C, HE = C_P). Note that differences between both categories of extended heuristics are marked by their names, using '_C' and '_C_P' as extensions, respectively.

In initial simulation experiments we found that MASP_AD may perform worse for high work loads. This is due to its greedy nature (van der Zee 2010), allowing for processing (many) small batches of jobs which require short processing times. Hence set-up efficiencies may be hurt. We found how a new truncation rule may mitigate these effects. According to the proposed truncation rule process batches for which set-up time divided by batch size is smaller than the difference between the mean job arrival interval and the mean job processing time at the batch stage $(\frac{s^B_{j_0,j}}{k} < (\bar{\lambda} - \bar{p}^B))$ are preferred. Note that the latter difference specifies machine capacity available for executing set-ups.

Three levels for the set-up to run-time ratio, i.e. mean set-up time divided by mean processing time required at the batch machine, are considered: 0.25, 0.50 and 1.00. Each alternative choice of set-up to run-time ratio is implemented by adjusting the set-up matrix, i.e. multiplying set-up times specified by the respective set-up to run-time ratio, see Table 3. Previous studies stress the high impact of the level of the set-up to run-time ratio on the heuristics' (relative) performance, see, for example, Mahmoodi and Dooley (1991), Ruben, Mosier, and Mahmoodi (1993), and Shambu, Suresh, and Pegels (1996), Pickardt and Branke (2012).

Alternative workload levels of 60, 75 and 90%, including both processing and set-ups, are chosen by adapting the mean inter-arrival time. The levels have been determined by adopting FCFAM as a benchmark for controlling the batch stage. Arrival rates have been established for each setting of the number of job families and the set-up to run-time ratio.

Wemmerlöv and Vakharia (1991), and Mahmoodi and Martin (1997) indicate that shop workload has a major impact on (relative) performance of family based dispatching heuristics.

The number of families ranges from 4 to 16. This range is in conformity with many other studies, see, for example, Jensen, Malhotra, and Philipoom (1996), Shambu, Suresh, and Pegels (1996), Marsh, Shafer, and Meredith (1999), Pickardt and Branke (2012), van der Zee (2015).

5.2 *Simulation modelling*

Plant Simulation^{TM} 10.1 (Siemens PLM Software 2016) is used to carry out the simulation experiments. A total of 60 runs is considered for each experiment. The length of the warm-up period is determined using the Welch procedure (Law 2015). In accordance with the outcomes of the procedure the warm up period and run length are set at 1,000,000 and 11,000,000 time units, respectively. SPSS 24.0 for Windows (IBM 2016) is used for performing Analysis of Variance (ANOVA) for analysing effects of alternative shop configurations on heuristics' performance.

6. Analysis of simulation results

In this section we will analyse the outcomes of the simulation study, and consider managerial implications of insights obtained. Table 4 shows mean job flow times for each heuristic across all shop configurations. Effects of alternative shop configurations on heuristics' performance outcomes have been studied using ANOVA, see Table 5 and Figures 2–5. All effects are significant ($p < 0.001$). To establish best performing heuristics for each shop configuration Tukey's HSD (Honest Significant Difference) test has been applied for each shop configuration. Outcomes for best performing heuristics are printed in bold, see Table 4.

Figure 2 indicates potential of the extended heuristics, by showing the way extensions influence study outcomes in terms of the marginal means of mean job flow times for all shop configurations studied. Outcomes indicate that shop performance may – on average – be improved by about 5%, also see Table 4. Pro-active set-ups contribute about 3% to this figure, compare Figure 2 (HE = C_P). Performance differences for alternative shop configurations range from 2 to 8%. Differences in gains reported may be largely explained by the set-up to run-time ratio, see below. Relative performance differences among heuristics seem to be hardly influenced by heuristics' extensions, see Figure 2. This is confirmed by Table 4 which indicates that the heuristics performance rankings for specific configurations are barely changed due to proposed extensions. Note that heuristic rankings confirm earlier findings (Pickardt and Branke 2012) by showing that there is no single best heuristic.

Effectiveness of shop coordination is most clearly expressed for high set-up to run-time ratios (Figure 3). A likely explanation for this finding is that realising set-up efficiencies matters most under these circumstances. The fact that under these circumstances exhaustive heuristics (MASP, MAS) – which seek to improve set-up efficiencies by maximising batch size – outperform non-exhaustive heuristics (MASP_AD, SNSPT) – which relax constraints on batch size, confirms this. Moreover, good performance of MAS relative to MASP – which extends the MAS information base by including job processing times, see (3) – stresses that in case of long set-up times a singular focus on set-up efficiencies is worthwhile.

Figures 4 and 5 provide further information on the way shop workload and the number of job families influence heuristics' performance. Figure 4 shows that an increase of shop workload does hardly impact relative gains realised by heuristics' extensions. It does, however, influence heuristics' performance rankings. Performance differences for better informed heuristics (MASP, SNSPT) vs. less informed heuristics (FCFAM, MAS) and greedy heuristics (MASP_AD) tend to increase for higher workloads. Likewise, the number of job families does hardly interact with proposed extensions for heuristics, see Figure 5. However, it does influence relative performance for individual heuristics. For higher numbers of job families performance differences among MASP, MASP_AD and SNSPT tend to diminish, whereas MAS and FCFAM perform worse. This can be explained by the fact that under these circumstances, available jobs in queue per family tend to be small, implying few possibilities for differentiation among process batch sizes. Hence, opportunities for realising set-up efficiencies are less, while relevance of job processing times in family priority setting increases (compare MAS vs. MASP, MASP_AD and SNSPT). At the same time, shorter queue lengths make the heuristics' nature – being exhaustive or non-exhaustive – of less importance (compare MASP, MASP_AD and SNSPT).

Concluding, proposed extensions for family-based heuristics may improve shop performance by up to 8%. Size of gains are in line with those found for related research (Mahmoodi and Martin 1997; Nomden, Van der Zee, and Slomp 2008; van der Zee, Gaalman, and Nomden 2011), see Section 2. Obviously, benefits of heuristics' implementation have

Table 4. Mean job flow times for existing and proposed heuristics.

Shop configuration			Existing heuristics					Proposed heuristicsShop coordination					Shop coordination & pro-active set-ups				
# Job Families	S/R	Work Load (%)	FCFAM	MASP	MAS	MAS_AD	SNSPT	FCFAM_C	MASP_C	MAS_C	MASP_AD_C	SNSPT_C	FCFAM_C_P	MASP_C_P	MAS_C_P	MASP_AD_C_P	SNSPT_C_P
4	0.25	60	793.02	782.08	793.04	**773.36**	779.36	786.07	781.83	780.84	**769.64**	**769.65**	772.73	**762.76**	770.80	**754.99**	**755.98**
4	0.25	75	764.76	740.73	761.56	724.31	**722.53**	756.89	732.44	747.81	**714.84**	**712.36**	738.62	717.03	735.36	**702.88**	**702.69**
4	0.25	90	1015.86	959.09	994.60	964.03	**874.33**	992.91	943.01	974.32	935.18	**860.40**	972.57	931.77	960.25	939.71	**851.18**
4	0.50	60	897.51	**884.86**	890.98	**882.12**	**879.50**	891.04	**874.15**	**882.87**	**874.07**	**872.35**	854.81	851.95	**848.20**	852.60	**839.17**
4	0.50	75	828.38	809.76	819.73	815.11	**801.26**	811.53	799.97	805.56	802.10	**791.48**	790.53	**773.87**	780.61	777.91	**767.43**
4	0.50	90	967.25	923.73	937.77	952.69	**909.25**	934.18	905.75	916.46	932.90	**891.40**	917.95	**882.83**	891.93	913.52	**880.10**
4	1.00	60	1092.45	1085.55	1090.87	1086.19	**1082.29**	**1080.79**	**1071.22**	**1078.09**	**1076.14**	**1066.71**	**1026.50**	**1018.14**	**1018.79**	**1027.10**	**1017.57**
4	1.00	75	982.97	**970.96**	**971.24**	979.35	**974.43**	969.40	**955.29**	**953.47**	967.39	**957.93**	921.21	**903.60**	**910.62**	920.93	**906.98**
4	1.00	90	1027.47	995.16	**988.01**	1053.56	1037.24	985.04	**956.50**	**961.22**	1019.81	1002.81	958.17	**929.91**	**931.63**	999.83	981.24
8	0.25	60	823.94	**799.61**	817.64	**795.19**	**799.93**	817.17	792.42	808.23	**789.04**	**787.92**	796.19	**778.10**	787.28	**773.82**	**774.29**
8	0.25	75	802.65	754.19	792.11	**744.05**	**746.88**	791.73	742.68	778.70	**732.42**	736.08	773.88	**727.53**	759.37	**720.00**	**721.03**
8	0.25	90	1050.57	945.56	1008.84	947.59	**902.48**	1025.52	927.54	977.63	937.41	**889.23**	1009.69	911.81	967.16	923.59	**874.28**
8	0.50	60	948.30	**931.14**	940.43	935.44	**929.20**	937.46	**922.33**	**931.84**	**922.06**	**920.72**	900.04	**887.94**	**895.63**	**886.20**	**885.27**
8	0.50	75	907.14	**857.62**	884.24	870.77	859.42	884.38	**845.85**	869.33	851.43	**848.37**	853.38	**816.10**	838.24	**823.27**	**815.69**
8	0.50	90	1057.04	**989.41**	1001.15	1023.16	**986.98**	1026.85	**953.10**	973.97	1005.23	969.95	998.78	**937.50**	950.01	987.02	955.84
8	1.00	60	1200.06	**1181.40**	1185.81	1184.19	**1174.13**	1179.13	**1167.71**	**1172.41**	**1163.64**	**1170.54**	1117.95	**1089.13**	**1101.51**	**1107.12**	**1099.37**
8	1.00	75	1113.49	**1073.28**	1087.01	1088.35	1079.99	1089.03	**1049.96**	1065.52	1071.64	1066.22	1027.04	**991.90**	1010.54	1011.83	**1004.29**
8	1.00	90	1193.39	1130.33	**1124.11**	1176.96	1188.38	1155.60	**1082.53**	**1091.80**	1150.19	1150.63	1115.93	**1057.70**	**1053.37**	1117.83	1120.97
16	0.25	60	833.48	**803.98**	829.32	**804.18**	**806.49**	826.68	**800.63**	822.74	**801.09**	**805.56**	803.29	781.96	798.80	**770.20**	782.07
16	0.25	75	838.29	766.10	824.24	**760.21**	768.25	823.64	**757.02**	808.47	**753.03**	**754.76**	803.27	**742.50**	793.19	**733.65**	**736.73**
16	0.25	90	1121.87	949.19	1046.31	975.16	**922.92**	1094.92	923.02	1022.40	965.24	**906.07**	1087.36	917.23	1007.03	951.71	**895.52**
16	0.50	60	982.71	**960.46**	972.15	**955.50**	**955.93**	973.72	**954.12**	**961.25**	**946.80**	**950.83**	929.49	**909.20**	923.49	**906.53**	**907.98**
16	0.50	75	944.67	**877.77**	916.98	**882.77**	**879.62**	924.46	**860.56**	903.94	**867.48**	**866.58**	890.83	**826.60**	863.21	**832.43**	**831.91**
16	0.50	90	1158.53	**1034.61**	1068.49	**1045.15**	**1037.07**	1127.57	**997.11**	1041.57	1034.18	1014.50	1095.31	**976.81**	1012.16	1010.86	994.19
16	1.00	60	1262.64	**1237.64**	1245.41	**1231.03**	**1232.12**	1253.90	**1226.75**	**1237.96**	**1222.66**	**1225.39**	**1161.95**	**1149.94**	**1153.47**	**1151.86**	**1151.89**
16	1.00	75	1195.49	**1135.56**	1165.45	1152.69	1147.10	1177.69	**1119.87**	1143.75	1137.74	**1125.83**	1095.41	**1057.99**	1074.59	**1069.78**	**1065.28**
16	1.00	90	1355.21	**1233.77**	1237.53	1247.62	1298.20	1311.49	**1192.70**	1208.82	1222.19	1250.58	1260.52	**1152.23**	**1161.60**	1186.76	1215.36
Overall			1005.89	956.06	977.59	964.84	954.64	986.25	938.37	960.04	950.58	939.44	950.87	906.82	925.88	920.52	908.68

Table 5. ANOVA results for mean job flow time.

Source	*F*	*p*-value	Source	*F*	*p*-value
H	2752.633	0.000	H * SR * WL	95.000	0.000
SR	328,949.973	0.000	H * SR * F	6.466	0.000
WL	86,829.219	0.000	H * WL * F	32.591	0.000
F	51,684.717	0.000	SR * WL * F	309.474	0.000
H * SR	346.661	0.000	H * SR * WL * F	2.757	0.000
H * WL	226.328	0.000			
H * F	132.262	0.000			
SR * WL	14,469.081	0.000			
SR * F	8858.582	0.000			
WL * F	1179.305	0.000			

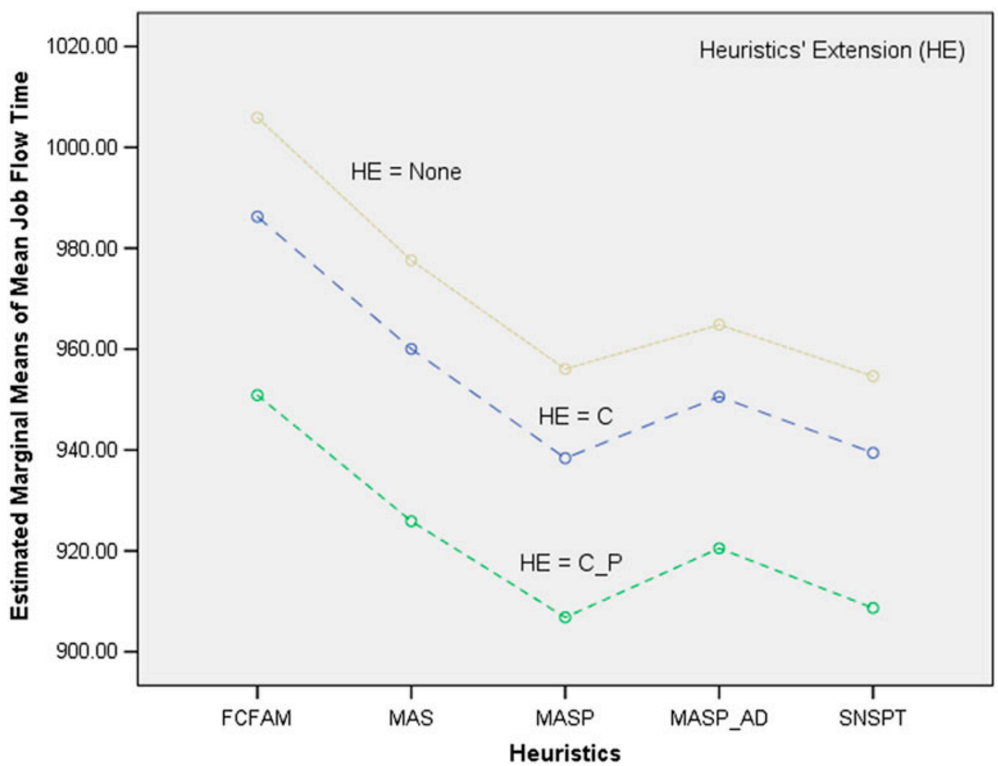

Figure 2. Estimated marginal means of mean job flow times for heuristics' extensions.

to be traded off against implementation costs, i.e. costs of data collection, set-up or reconfiguration of the shop floor control system, and planner/operator education. Simulation outcomes offer some guidance in making this trade-off, by suggesting to include the set-up to run-time factor in decision-making – as it explains most of the gains of heuristics' extensions. On the other hand, the fact that choice of heuristics, shop workload and number of families hardly influence gains reported, suggest robustness of the solution for future changes in the shop environment, once implemented. Furthermore, intuitiveness and simplicity of the decision logic underlying heuristics are assumed to reduce costs of their implementation.

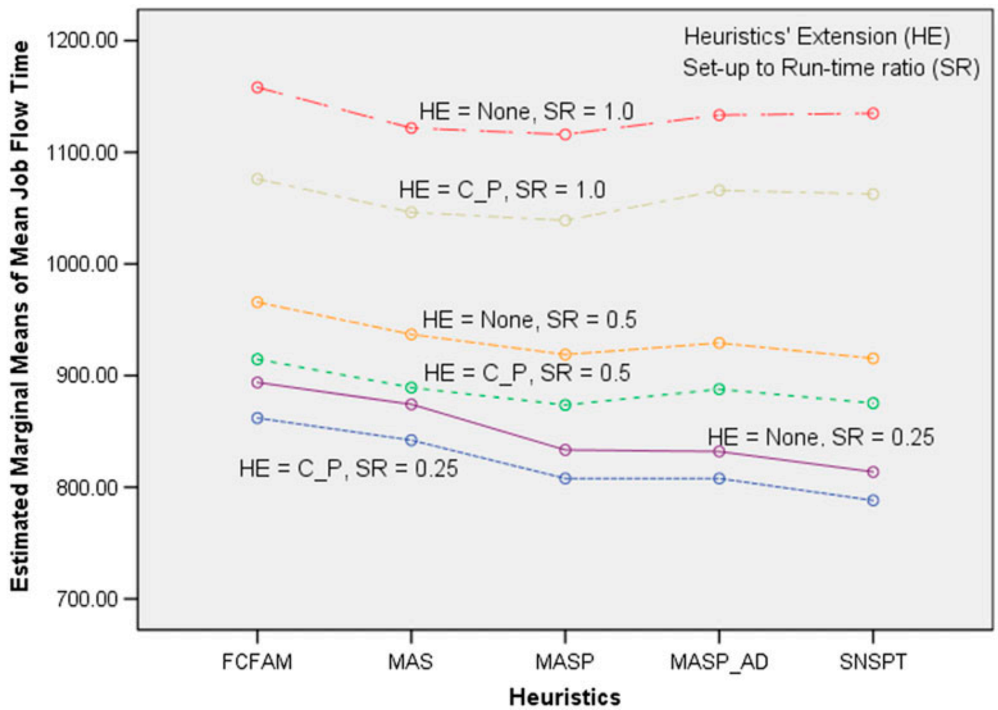

Figure 3. Estimated marginal means of mean job flow times for heuristics' extensions – alternative settings of set-up to run-time ratio.

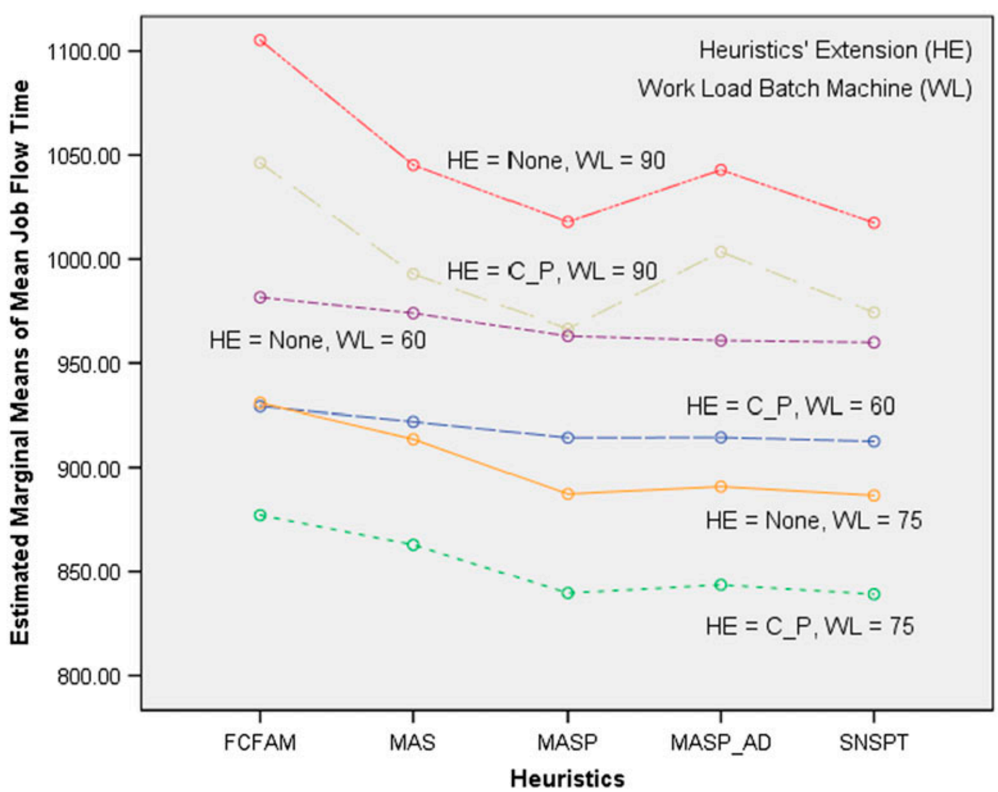

Figure 4. Estimated marginal means of mean job flow times for heuristics' extensions – alternative work load settings for batch machine.

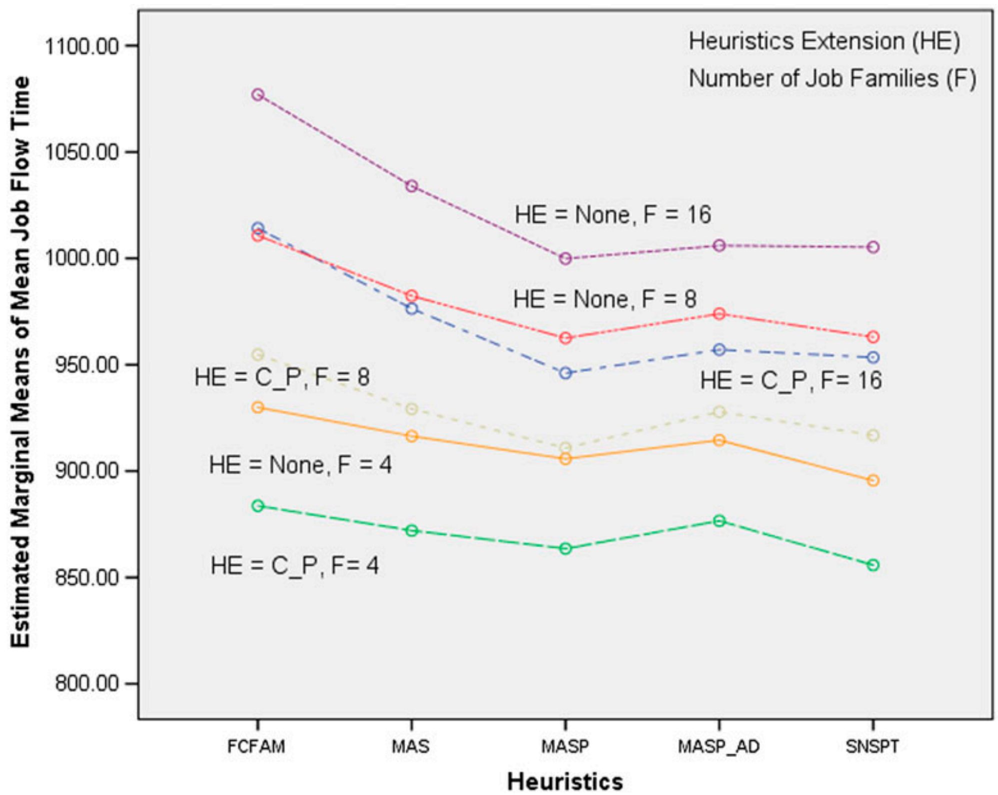

Figure 5. Estimated marginal means of mean job flow times for heuristics' extensions – alternative settings of number of job families.

7. Concluding remarks

Motivated by industrial case examples, in this article, we propose family-based dispatching heuristics for shop control that coordinate batching decisions with upstream stages for improving set-up efficiencies. Whereas most existing heuristics restrict focus to the batch stage, proposed heuristics extend decision scope by considering possibilities for re-sequencing jobs upstream and initiating set-ups in anticipation of jobs yet to arrive. An extensive simulation study concerning a two-stage flow shop indicates that shop performance may be improved by 2–8% by implementing extended heuristics.

Analysis of simulation outcomes showed that relative performance rankings for heuristics studied (FCFAM, MAS, MASP, MASP_AD, SNSPT) are hardly influenced by proposed extensions. Magnitude of shop performance improvements is mainly influenced by the set-up to run-time ratio. Whereas a low set-up to run-time ratio (0.25) is related to improvements of about 2%, the presence of a high set-up to run-time ratio (1.0) allows extended heuristics to outperform existing heuristics by around 8%. Workload levels and number of job families appeared to have little effect on these figures.

Similar to common dispatching rules proposed heuristics support prompt decision-making. Furthermore, the simplicity, and intuitiveness of their decision logic are assumed to contribute to their uptake in practice.

Many avenues for further research on the coordination of batching decisions with upstream and downstream stages may be considered, involving various shop configurations, as determined by, for example, characteristics of the set-up process (sequence dependent set-ups vs. sequence independent set-ups, alternative set-up matrices), machine characteristics at the batch stage (single machine vs. parallel machines), or job routing (flow shops vs. job shops), or number of stages. Furthermore, we would like to mention recent research by Heger et al. (2016) who show how dispatching rules may be dynamically adjusted to shop conditions by changing their parameters. Their approach may be helpful in addressing the fact that there is no family based heuristic outperforming across various shop configurations. Likewise, one may wonder about the benefits of employing algorithms focussing at further improvement of the shop schedule, such as, for example, local search procedures (Ciavotta, Meloni, and Pranzo 2009; Pinedo 2015).

Disclosure statement

No potential conflict of interest was reported by the author.

References

Allahverdi, A. 2015. “The Third Comprehensive Survey on Scheduling Problems with Setup times/Costs.” *European Journal of Operational Research* 246: 345–378.

Bevilacqua, M., F. E. Ciarapica, and I. De Sanctis. 2017. “Lean Practices Implementation and Their Relationships with Operational Responsiveness and Company Performance: An Italian Study.” *International Journal of Production Research* 55 (3): 769–794.

Blackstone, J. H., D. T. Phillips, and G. L. Hogg. 1982. “A State-of-the-art Survey of Dispatching Rules for Manufacturing Job Shop Operations.” *International Journal of Production Research* 20 (1): 27–45.

Bokhorst, J. A. C., G. Nomden, and J. Slomp. 2008. “Performance Evaluation of Family-based Dispatching in Small Manufacturing Cells.” *International Journal of Production Research* 46 (22): 6305–6321.

Chern, C. C., and Y. L. Liu. 2003. “Family-based Scheduling Rules of a Sequence-dependent Wafer Fabrication System.” *IEEE Transactions on Semiconductor Manufacturing* 16 (1): 15–25.

Ciavotta, M., C. Meloni, and M. Pranzo. 2009. “Scheduling Dispensing and Counting in Secondary Pharmaceutical Manufacturing.” *AIChE Journal* 55 (5): 1161–1170.

Ciavotta, M., C. Meloni, and M. Pranzo. 2013. “Minimising General Setup Costs in a Two-stage Production System.” *International Journal of Production Research* 51 (8): 2268–2280.

Flynn, B. B. 1987. “Repetitive Lots: The Use of a Sequence-dependent Set-up Time Scheduling Procedure in Group Technology and Traditional Shops.” *Journal of Operations Management* 7 (1–2): 203–216.

Frazier, G. V. 1996. “An Evaluation of Group Scheduling Heuristics in a Flow-line Manufacturing Cell.” *International Journal of Production Research* 34 (4): 959–976.

Gavett, J. W. 1965. “Three Heuristic Rules for Sequencing Jobs to a Single Production Facility.” *Management Science* 11 (8): B-166–B-176.

Groover, M. P. 2008. *Automation, Production Systems, and Computer-integrated Manufacturing*. 3rd ed. New York: Prentice Hall.

Heger, J., J. Branke, T. Hildebrandt, and B. Scholz-Reiter. 2016. “Dynamic Adjustement of Dispatching Rule Parameters in Flow Shops with Sequence Dependent Set-up times.” *International Journal of Production Research* (on-line).

Hopp, W. J., and M. L. Spearman. 2008. *Factory Physics*. 3rd ed. Boston, MA: McGrawHill.

IBM. 2016. *SPSS 24.0 for Windows Software Package*, Chicago, IL.

Jayamohan, M. S., and C. Rajendran. 2000. “New Dispatching Rules for Shop Scheduling: A Step Forward.” *International Journal of Production Research* 38 (3): 563–586.

Jensen, J. B., M. K. Malhotra, and P. R. Philipoom. 1996. “Machine Dedication and Process Flexibility in a Group Technology Environment.” *Journal of Operations Management* 14 (1): 19–39.

Jensen, J. B., M. K. Malhotra, and P. R. Philipoom. 1998. “Family-based Scheduling of Shops with Functional Layouts.” *International Journal of Production Research* 36 (10): 2687–2700.

Kannan, V. R. 1998. “Analysing the Trade-off between Efficiency and Flexibility in Cellular Manufacturing Systems.” *Production Planning & Control* 9 (6): 572–579.

Kannan, V. R., and S. Ghosh. 1996. “Cellular Manufacturing Using Virtual Cells.” *International Journal of Operations and Production Management* 16: 99–112.

Kochhar, S., and R. J. T. Morris. 1987. “Heuristic Methods for Flexible Flow Line Scheduling.” *Journal of Manufacturing Systems* 6 (4): 299–314.

Law, A. M. 2015. *Simulation Modeling and Analysis*. 5th ed. Boston, MA: McGraw-Hill.

Li, S. 1997. “A Hybrid Two-stage Flowshop with Part Family, Batch Production, Major and Minor Set-ups.” *European Journal of Operational Research* 102 (1): 142–156.

Mahmoodi, F., and K. J. Dooley. 1991. “A Comparison of Exhaustive and Non-exhaustive Group Scheduling Heuristics in a Manufacturing Cell.” *International Journal of Production Research* 29 (9): 1923–1939.

Mahmoodi, F., and G. E. Martin. 1997. “A Newshop-based and Predictive Scheduling Heuristic for Cellular Manufacturing.” *International Journal of Production Research* 35 (2): 313–326.

Mahmoodi, F., K. J. Dooley, and P. J. Starr. 1990. “An Investigation of Dynamic Group Scheduling Heuristics in a Job Shop Manufacturing Cell.” *International Journal of Production Research* 28 (9): 1695–1711.

Marsh, R. F., S. M. Shafer, and J. R. Meredith. 1999. “A Comparison of Cellular Manufacturing Research Presumptions with Practice.” *International Journal of Production Research* 37 (14): 3119–3138.

McKay, K. M., F. R. Safayeni, and J. A. Buzacott. 1988. “Job-shop Scheduling Theory: What is Relevant?” *Interfaces* 18 (4): 84–90.

Mizrak, P., and G. Bayhan. 2006. “Comparative Study of Dispatching Rules in a Real-life Job Shop Environment.” *Applied Artificial Intelligence* 20: 585–607.

Mosier, C. T., D. A. Elvers, and D. Kelly. 1984. “Analysis of Group Technology Scheduling Heuristics.” *International Journal of Production Research* 22 (5): 857–875.

Neufeld, J. S., J. N. D. Gupta, and U. Buscher. 2016. “A Comprehensive Review of Flowshop Group Scheduling Literature.” *Computers & Operations Research* 70: 56–74.

Nomden, G. 2011. “Virtual Cellular Manufacturing – Relevance and Development of Heuristics for Family-based Dispatching.” PhD thesis, University of Groningen.

Nomden, G., and D. J. van der Zee. 2008. “Virtual Cellular Manufacturing: Configuring Routing Flexibility.” *International Journal of Production Economics* 112 (1): 439–451.

Nomden, G., J. Slomp, and N. C. Suresh. 2006. “Virtual Manufacturing Cells: A Taxonomy of past Research and Identification of Future Research Issues.” *International Journal of Flexible Manufacturing Systems* 17 (2): 71–92.

Nomden, G., D. J. Van der Zee, and J. Slomp. 2008. “Family-based Dispatching: Anticipating Future Jobs.” *International Journal of Production Research* 46 (1): 73–97.

Oeulhadj, D., and S. Petrovic. 2009. “A Survey of Dynamic Scheduling in Manufacturing Systems.” *Journal of Scheduling* 12: 417–431.

Pickardt, C. W., and J. Branke. 2012. “Setup-oriented Dispatching Rules – A Survey.” *International Journal of Production Research* 55 (20): 5823–5842.

Pinedo, M. 2015. *Scheduling: Theory, Algorithms, and Systems*. 5th ed. Cham Heidelberg: Springer.

Ponnambalam, S. G., P. Aravindan, and K. R. R. Reddy. 1999. “Analysis of Group-scheduling Heuristics in a Manufacturing Cell.” *The International Journal of Advanced Manufacturing Technology* 15 (12): 914–932.

Reddy, V., and T. T. Narendran. 2003. “Heuristics and Sequence-dependent Set-up Jobs in Flow Line Cells.” *International Journal of Production Research* 41 (1): 193–206.

Ruben, R. A., C. T. Mosier, and F. Mahmoodi. 1993. “A Comprehensive Analysis of Group Scheduling Heuristics in a Job Shop Cell.” *International Journal of Production Research* 31 (6): 1343–1369.

Russell, G. R., and P. R. Philipoom. 1991. “Sequencing Rules and Due Date Setting Procedures in Flow Line Cells with Family Setups.” *Journal of Operations Management* 10 (4): 524–545.

Salmasi, N., R. Logendran, and M. R. Skandari. 2010. “Total Flow Time Minimization in a Flowshop Sequence-dependent Group Scheduling Problem.” *Computers & Operations Research* 37: 199–212.

Sarin, S. C., A. Varadarajan, and L. Wang. 2011. “A Survey of Dispatching Rules for Operational Control in Wafer Fabrication.” *Production Planning & Control* 22 (1): 4–24.

Sels, V., N. Gheysen, and M. Vanhoucke. 2012. “A Comparison of Priority Rules for the Job Shop Scheduling Problem under Different Flow Time- and Tardiness-related Objective Functions.” *International Journal of Production Research* 50 (15): 4255–4270.

Shambu, G., N. C. Suresh, and C. C. Pegels. 1996. “Performance Evaluation of Cellular Manufacturing Systems: A Taxonomy and Review of Research.” *International Journal of Operations and Production Management* 16 (8): 81–103.

Siemens PLM Software. 2016. *Plant Simulation^{TM} 10.1*. Stuttgart.

Stuber, F. 1998. “Approaches to Shop Floor Scheduling – A Critical Review.” In: *Shop Floor Control – A Systems Perspective*, edited by E. Scherer. Berlin: Springer.

van der Zee, D. J. 2010. “Non-exhaustive Family Based Dispatching Heuristics–Exploiting Variances of Processing and Set-up Times.” *International Journal of Production Research* 48 (13): 3783–3802.

van der Zee, D. J. 2015. “Family-based Dispatching with Parallel Machines.” *International Journal of Production Research* 53 (19): 5837–5856.

van der Zee, D. J., G. J. C. Gaalman, and G. Nomden. 2011. “Family Based Dispatching in Manufacturing Networks.” *International Journal of Production Research* 49 (23): 7059–7084.

Wemmerlöv, U. 1992. “Fundamental Insights into Part Family Scheduling: The Single Machine Case.” *Decision Sciences* 23 (3): 565–595.

Wemmerlöv, U., and A. J. Vakharia. 1991. “Job and Family Scheduling of a Flow-line Manufacturing Cell: A Simulation Study.” *IIE Transactions* 23 (4): 383–393.

Wilson, A. D., R. E. King, and T. J. Hodgson. 2004. “Scheduling Non-similar Groups on a Flow Line: Multiple Group Setups.” *Robotics and Computer-integrated Manufacturing* 20: 505–515.

Appendix 1

Notation

Indexes

i, c	job identifier = 1, 2, … for jobs in the system
j	family identifier = 1, 2, …
k	batch size, i.e. number of jobs included in the batch

Parameters

c^*	highest priority job at serial stage
d_c	moment job c will be dispatched at the batch stage
j_0	current family, i.e. the family for which the batch machine has been set-up
j^*	highest priority family (in family selection)
$p_{i,j}^B$	processing time of job i = 1, 2, … belonging to family j for batch stage
$p_{i,j}^S$	processing time of job i = 1, 2, … belonging to family j for serial stage
$\bar{p}^B$	mean processing at the batch stage over all job families
$\hat{p}$	mean processing time for all jobs in queue at the batch stage
q_j^B	number of jobs in queue for family j at t_0 at the batch stage
q_j^S	number of jobs in queue for family j at t_0 at the serial stage
$s_{j_0,j}^B$	set-up time required for family j at batch stage, given current family j_0
$\tilde{s}_{j_0,j}^B$	adjusted set-up time required for family j at batch stage, given current family j_0 (pro-active set-ups allowed)
$\hat{s}$	mean set-up time for all jobs in queue at the batch stage
t_0^B	decision moment for the batch stage, i.e. the moment the dispatcher is triggered to make a decision
t_0^S	decision moment for the serial stage, i.e. the moment the dispatcher is triggered to make a decision
$t_{i,j}^B$	arrival moment at batch stage of job i = 1, 2, … belonging to family j
C	set of jobs that candidate for dispatching at serial stage (family selection)
$I_j^B(t)$	set of jobs in queue for family j at the batch stage
$I_j^S(t)$	set of jobs in queue for family j at the serial stage
J	set of job families
N	total number of jobs processed over all families
Q	parameter SNSPT rule (Kochhar and Morris 1987)
S_c	schedule for batch stage, assuming job c to be processed at the serial stage
$\bar{S}$	set of schedules S_c
S^*	preferred schedule for the batch stage (family selection)
T^B	moment the batch machine starves
α, β, γ	parameters SNSPT rule (Kochhar and Morris 1987)
$\bar{\lambda}$	shop arrival rate, $\bar{\lambda} = \sum_{j \in J} \lambda_j$

Variables

$d(S_c, c)$	moment job c will be dispatched at the batch stage according to schedule S_c
$ft_{i,j}$	flow time for job i = 1, 2, … belonging to family j
$st^B(c)$	first possible moment job c (present in queue at the serial stage) may be processed at the batch stage
$w_{i,j}^B$	waiting time for job i = 1, 2, … belonging to family j at batch stage
$w_{i,j}^S$	waiting time for job i = 1, 2, … belonging to family j at serial stage
$\bar{T}$	mean flow time per job

Appendix 2

Table A1. Overview of family based dispatching heuristics.

Name	Explanation[a]	Reference	Included[b] in study	Extensions considered[c]
FCFAM	First Come FAMily	Flynn (1987)	Yes	FCFAM_C, FCFAM_C_P
LPTMM		Mahmoodi and Martin (1997)	No	
MAS	Minimum Average Set-up time	Nomden, Van der Zee, and Slomp (2008)	Yes	MAS_C, MAS_C_P
MASP	Minimum Average Set-up plus Processing time	Jensen, Malhotra, and Philipoom (1996)	Yes	MASP_C, MASP_C_P
MASP_AD	ADaptive Minimum Average Set-up plus Processing time	van der Zee (2010)	Yes	MASP_AD_C, MASP_AD_C_P
MASP_HY	HYbrid Minimum Average Set-up plus Processing time	van der Zee (2010)	No	
PH	Proposed Heuristic	Reddy and Narendran (2003)	No	
SNSPT	Shortest Normalised Setup and Processing Time	Kochhar and Morris (1987)	Yes	SNSPT_C, SNSPT_C_P
SST	Short Set-up Time	Gavett (1965)	No	
VCM 3	Virtual Cellular Manufacturing 3	Kannan and Ghosh (1996)	No	
VCM 5	Virtual Cellular Manufacturing 5	Kannan and Ghosh (1996)	No	

[a]Each heuristic is referred to by its acronym. If an explanation of the acronym is available in literature it is mentioned here.
[b]Indicates whether a heuristic is included in the simulation study or not. Existing heuristics that showed good performance in previous studies have been included in the study.
[c]Extended heuristics mentioned build on existing heuristics. To isolate and assess benefits of considering pro-active set-ups in family-based dispatching for shop performance we distinguish among extended heuristics not acknowledging pro-active set-ups (C) and those that do (C_P).